A Secure Storage Scheme for Healthcare Data Over the Cloud Based on Multiple Authorizations

Zeyad A. Al-Odat*, Sudarshan K. Srinivasan*, Eman M. Al-Qtiemat*, Sana Shuja†
*Electrical and Computer Engineering, North Dakota State University
Fargo, ND, USA
†Electrical Engineering, COMSATS Institute of Information Technology,
Islambad, Pakistan
Emails: *zeyad.alodat@ndsu.edu, *sudarshan.srinivasan@ndsu.edu, *eman.alqtiemat@ndsu.edu,
†SanaShuja@comsats.edu.pk

***Abstract*—This paper introduces a secure storage scheme for healthcare data with multiple authorizations. The proposed design divides the healthcare data into small parts and distributes them over multiple cloud locations. The Shamir's Secret Sharing and Secure Hash Algorithm are employed to provide the security and authenticity requirements to the proposed design. The design comprises two phases, the distribution phase, and the retrieving phase. The distribution phase comprises three operations of dividing, encrypting, and distribution. The retrieving phase performs collecting and verifying operations. To increase the security level, the encryption key is divided into secret shares using Shamir's Secret Sharing. Moreover, the Secure Hash Algorithm is used to verify the healthcare data after retrieving from the cloud. The experimental results show that the proposed design can reconstruct a distributed healthcare data with a significant speed while conserving the security and authenticity properties.**

***Keywords*—*Healthcare*; *Security*; *Shamir's Secret Sharing*; *Authorization*.**

I. Introduction

The global healthcare system is changing every day, particularly the conversion into a digital healthcare environment that contains all patient's data and their corresponding records [1]. This change is stimulated by the new technologies, increased number of populations, and the change in people's lifestyles. The emerging technology offers a convenient environment for supporting healthcare management, digital clinics, monitoring, and preserving of health records [2].

Cloud computing is an emerging technology that is updated frequently and adopted with new technologies rapidly [3]. The big healthcare data is considered as a big data application over cloud computing. Therefore, all challenges, analyses, and concerns that are related to cloud computing are applied to the big healthcare data [4].

The security and privacy of big data are important because numerous amount of data is stored at the same pool of storage locations [5]. The security and privacy concerns of the healthcare data over the cloud are increasing, in addition to the concerns of healthcare institutions that found the security and privacy requirements of the healthcare data over the cloud are not adequate [2][6].

Big data security is guaranteed by different technologies, including the following: 1) Encryption, 2) Centralized key management, 3) User access control, 4) Intrusion detection and prevention, and 5) Physical security. Moreover, everyone is responsible for security, e.g., policies, agreement list, and security software. The security requirements of physical components are guaranteed by the Cloud Service Provider (CSP), which grants proper accesses to the data owners [7].

One of the main methods to share big data is distribution technology. The big data is divided into parts and distributed over several storage locations [8]. However, many criteria need to be addressed in this scheme including data security and retrieval. The data need to be secured against unauthorized access and protected from data tampering and alteration. Data security is achieved using encryption techniques, e.g., Advanced Encryption Standard (AES) while data integrity is achieved using the Secure Hash Algorithm (SHA). However, an attacker can hack the encryption key and access the big data. In the case of healthcare data, the attacker will gain access to the patients' sensitive information, e.g., social security numbers [9].

In this paper, we introduce a secure and authentic healthcare storage scheme over the cloud. In our design, the Shamir's Secret Sharing (SSS) and SHA are employed to provide the security requirements of the proposed scheme. The SSS is used to divide the encryption key into parts and distributes these parts over authorized entities. The $SHA-512$ is used to check the data integrity after retrieval [10].

The rest of paper is organized as follows: Section II provides a background information about the secure hash algorithm and Shamir's Secret Sharing; a literature review is presented in Section III; Sections IV exhibits the proposed methodology; results and discussions are conferred in Section V; Section VI concludes the paper.

II. Background

Before going through the details of our proposal, brief descriptions about The *SSS* and *SHA* will be presented in the subsequent text.

 ISBN: 978-1-61208-743-6

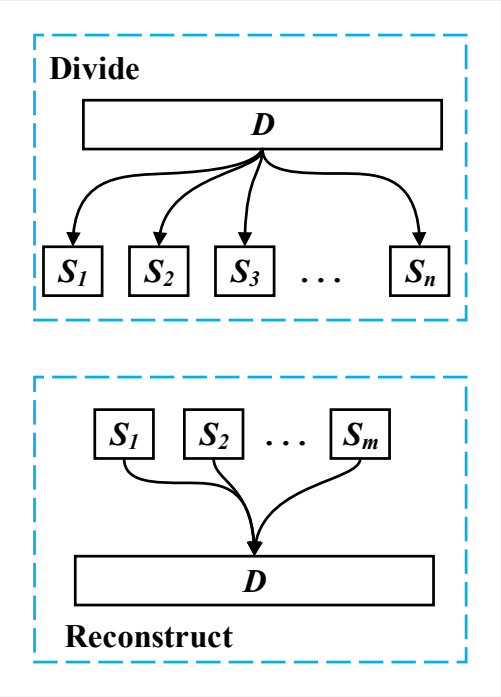

Figure 1. Shamir's Secret Sharing structure

A. Shamir's Secret Sharing

The SSS is a secret sharing technique that is proposed by Adi Shamir [11]. The *SSS* divides data (D) into a number of pieces (S_n) where D is easily reconstructable from the minimum number of pieces (S_m). The SSS is a (m, n) based scheme, where m is the minimum number of pieces that are needed to reconstruct the data (D) and n is the total number of pieces that the data (D) is divided to. Additionally, D is completely undetermined if the number of known pieces is fewer than $m - 1$, which means that m pieces must be available to reconstruct the original data D. Figure 1 shows the general structure of the SSS algorithm, where it involves two operations, the divide and reconstruct. The upper part of the figure shows that the data D is divided into n pieces. Then, to reconstruct the data D from these parts a minimum of m pieces is needed. More details will be presented in Section IV.

B. Secure Hash Algorithm

The SHA is a cryptography function that is used for integrity scrutiny. The SHA takes a message (M) of arbitrary size, then through compression function calculations produces the message hash (H). The SHA is used to provide the authenticity and integrity of the data, i.e., ensure that the data have not tampered during transmission or storing.

The secure hash algorithms follow two construction models. The first construction model is Merkle Damgard (*MD*), which is used to construct the hash functions *MD4*, *MD5*, *SHA-1*, and *SHA-2* [12]. The second one is the Sponge structure model that is used to construct the $SHA - 3$ hash function [13]. In our proposal, we use the MD structure model to provide data integrity and authenticity. Figure 2 shows the general structure of the MD structure model. The message M of size $< 2^{128}$ is preprocessed first by padding the input message to make its size a multiple of the block size (B). Then the padded message M^* is divided into equal size blocks (B_n).

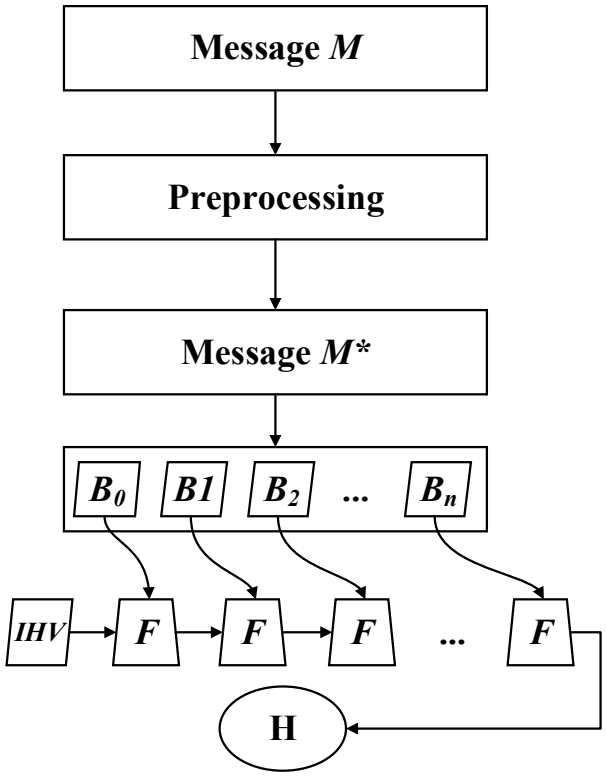

Figure 2. General structure of the Secure Hash Algorithm (SHA)

In the *MD* hash standards, the maximum message size that each algorithm accepts is dependent on the block size, where the 512-bit block size accepts messages of size less than 2^{64}-bit, while the others accept a message size of 2^{128}-bit. All hash standards perform the following steps:

1) **Message padding**. In this phase, the message is padded with a sufficient number of zeros to make the message size multiple of the block size.
2) **Message divide**. In this phase, the message is divided into equal size blocks (B), where the block size is dependent to the desired hash function.
3) **Compression function calculation**. All blocks are processed sequentially using compression function (F). The Initial Hash Value (IHV) is used to process the first block (B_0), then the output of processing each block is used as (IHV) to the next block calculation.
4) **Output hash generation**. The output of the last block calculation is taken as the output hash (H).

For more details about secure hash algorithms and their compression functions, the reader is referred to [14].

 ISBN: 978-1-61208-743-6

C. Threat Model

Two major threats are related to this work.

1) Privacy Threat. The healthcare data owners (patients, medical institutions, authorized entities) have concerns about the privacy of their data. These concerns include the encryption key exposure by an adversary and the *CSP* threat. Once the attacker gains access to the encryption key, all sensitive healthcare information will be exposed. Furthermore, the *CSP* can access the uploaded data and distributes them without the knowledge of the data owners. Therefore, the level of trust between the data owners and *CSPs* is reduced [15].
2) Integrity Threat. The healthcare data might be exposed to intentional or unintentional data tampering. The data owners are not aware of their uploaded data over the cloud and they consider that the *CSP* will take full responsibility for the uploaded data. However, the stored healthcare records may be tampered by an external adversary or due to software or hardware failures of the cloud service [16].

In our work, we preserve the healthcare data privacy over the cloud and provide a scheme that increases the level of trust between the data owners and the *CSP*.

III. Related Work

A secure leakage resilient s-health system has been proposed to protect the privacy keys from seizing that might happen because of some leakage attacks [17]. This approach presented a new cryptographic public key called leakage-resilient anonymous Hierarchical Identity-Based Encryption (HIBE). The security of this construction has been proven against chosen-plaintext attacks. Also, performance analysis has been done to demonstrate the practicability of this technique.

To solve security challenges, many approaches have been suggested based on Attribute-Based Encryption. Charanya *et al*. employed multiple parties in cloud computing to assure that eHealth data are secure [18]. Attributes and key policy have been used to encrypt health data, the only one who has this information can decrypt the health data. Decryption can be done only after verifying the attributes and key policy by using both a key distribution center and the secure data distributor. However, both of the aforementioned approaches consider the key privacy not all health data privacy.

An efficient data-sharing scheme called MedChain has been presented to control efficiency issues in the existing approaches such as sharing data streams, which are generated by monitoring devices [19]. This technique collects blockchain, digest chain, and structured peer to peer network techniques for sharing both types of healthcare data. For flexible data sharing, a session-based healthcare data-sharing scheme is designed. an evaluation process has been applied on MedChain illustrated that it can fulfill higher adequacy and satisfy the security-based requirements in data sharing.

A novel framework has been proposed for control accessing the Personal Health Records (PHRs) in the cloud computing environment [20]. To enable fine-grained and scalable access control for PHRs, Attribute-Based Encryption (ABE) techniques have been utilized to encrypt the PHRs for patients. Authors divided their system into various security domains, each domain is responsible for a subset of the users. Each patient has full control over his data, and the complexity of key management is reduced significantly. The proposed scheme is flexible, it supports efficient and on-demand repeal of user access rights.

Chen *et al*. have presented a secure dynamic access structure that can guarantee accurate access to the cloud server's medical records with multi-user settings [21]. Cryptography based on Lagrange multipliers has been used for encrypting the records to assure the maximum control of patients over their medical data. Mainly, this work focused on improving the encryption of PHRs and enhancing user dynamic access rules. This approach is very flexible in case of multi-user access and addition or modification of PHR.

The delay time of accessing the patient record can cause a death toll and decrease the health service level delivered by the medicinal professionals. A new study has been introduced that combines the Triple Data Encryption Standard (3DE) and Least Significant Bit (LSB) to enhance security measure that is applied to the patient's data [22]. For the experiment, a simulation program has been developed by using Java programming language. The experiment shows that patient's data and is saved, shared, and controlled in a secure way using the proposed combined method.

A new approach has been introduced to protect the identity and the privacy of the medical data using an effective encryption technique [23]. Also, an authorization framework has been discussed to control the access control mechanism of medical data. Encryption was done by using ARCANA that provides hierarchically access to many data resources. The XACML access model has been employed to subedit the access control framework. The AT&T scheme has been implemented to control the access mechanism of the patient's health data.

In the subsequent section, a secure healthcare storage and sharing scheme based on multiple authorizations will be presented.

IV. Proposed Methodology

The proposed design comprises two phases, distribution and retrieving. In the distribution phase, the healthcare data are signed, encrypted and distributed over multiple cloud locations. Also, the encryption key (E) is divided into parts and distributed over n authorized entities (*AEs*). In the retrieving phase, the healthcare data are collected from the storage locations using the (*SE*), and least number of authorized entities are requested to provide their secret part (E_n). The proposed design is implemented with aligning to Figure 3 and

 ISBN: 978-1-61208-743-6

Figure 5. To better understand the figures and the details of the proposed work, please refer to Table I that shows the used notations in this section.

TABLE I. NOTATIONS

Symbols	Meaning
D	The healthcare Data
SHA	Secure Hash Algorithm (*SHA-512*)
H	Hash value after applying the *SHA-512*
H^*	Hash value after retrieving D
SSS	Shamir's Secret Sharing
D_i	i part of Data D
E	Encryption Key
E_n	n parts of Encryption key E
$L(x)$	Lagrange polynomial to find $L(0)$
CSP	Cloud Service Provider
AE	Authorized Entity
SE	Service Entity that responsible for Data collection

A. Distribution Phase

In the distribution phase, the *SHA-512* compression function is applied to the healthcare data file. Then, the calculated hash value (H) is appended to the healthcare data (D), as shown in Figure 3. Afterward, the healthcare data file is divided into smaller blocks (D_1, D_2, ..., D_i), where the hash value (H) is appended to the last block (D_i). Each of the healthcare data parts is encrypted using the encryption key (E) and distributed over multiple cloud locations for storage and sharing.

The Encryption key (E) plays a major rule in the security of the healthcare data D. Therefore, the encryption key (E) is divided into shares (E_1, E_2, ..., E_n) using the SSS algorithm. Then, these shares are distributed over n authorized entities (AEs), where m number of shares must be present to calculate the original encryption key. According to the SSS algorithm, the number of shares (m) that are needed to reconstruct E is represented by a polynomial of power ($m-1$), as shown in Figure 4. The figure shows the pseudo-code of the general procedure to divide the encryption key (E) into shares (E_n). Firstly, the minimum number of shares (m) that are needed to reconstruct E is determined. Then, an (m-1) random numbers (a_{m-1}) are generated to construct the polynomial equation ($f(x)$) that is used to generate the required shares (n). Noting that, the value of a_0 is equal to the value of E. Afterward, the total number of shares (n) is determined, and n pairs of shares $(t, f(t))$ are generated using the polynomial function ($f(x)$). The generated secret shares are distributed over several authorized entities (*AEs*) including patients, physicians, medical institutions, and any other authorized entities, which are determined ahead of the secret shares generation process.

B. Retrieving Phase

In the cloud, the healthcare data parts are distributed over multiple storage locations. To keep track of all parts a unique identifier is given to each part after partitioning. In our proposal, one of the authorized entities, called Service Entity (*SE*), is responsible for data parts collection and parsing. The *AEs* have the correct order of the data parts and their locations

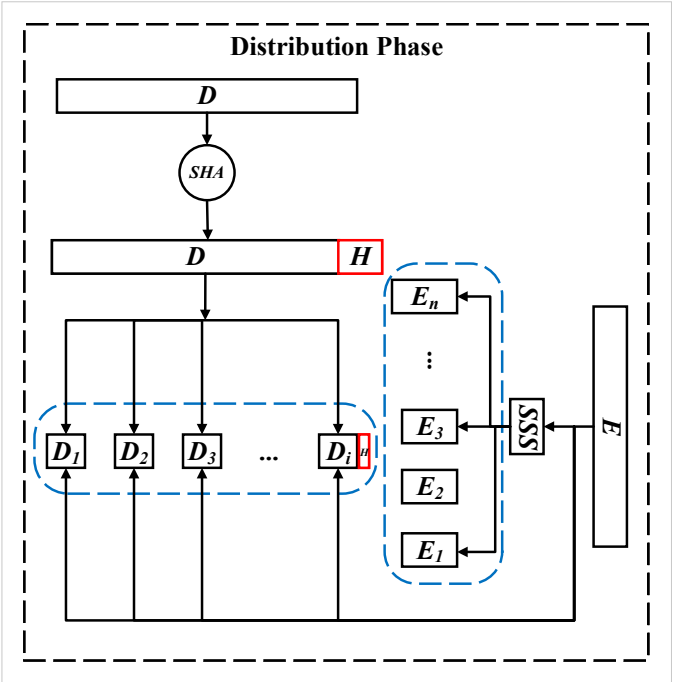

Figure 3. Schematic diagram of the Distribution phase

Algorithm 1: SSS

Input: Encryption Key (E)
Output: E_1, E_2, ..., E_n

$Determine(m)$; `//Least number of shares`
for $i \leftarrow 1$ ***to*** $m-1$ **do**
  $a_i = Rand()$
$f(x) = a_0 + a_1 x + a_2 x^2 + ... + a_{m-1} x^{m-1}$
$Determine(n)$; `//Total number of shares`
for $t \leftarrow 1$ ***to*** n **do**
  $E_t = (t, f(t))$

Figure 4. Pseudo Code of the SSS algorithm

over the cloud, and every time a retrieval job is requested the *SE* duty is assigned to one of the *AEs*. However, the *SE* is unable to decrypt the healthcare data because of the multiple authorization scheme that we deployed using the *SSS* algorithm. To ensure that the *SE* is not considered as a weak part in the design, the *SE* duty is assigned to one of the data owners every time a data retrieval request is performed.

To retrieve the healthcare data, one of the authorized entities sends an access request to the service entity. The service entity sends a secret-key share request to all authorized entities (AEs). According to the predefined configurations in the distribution phase, m number of secret shares must be provided to reconstruct the encryption/decryption key. Once collected, the encryption/decryption key (E) is computed using Lagrange

 ISBN: 978-1-61208-743-6

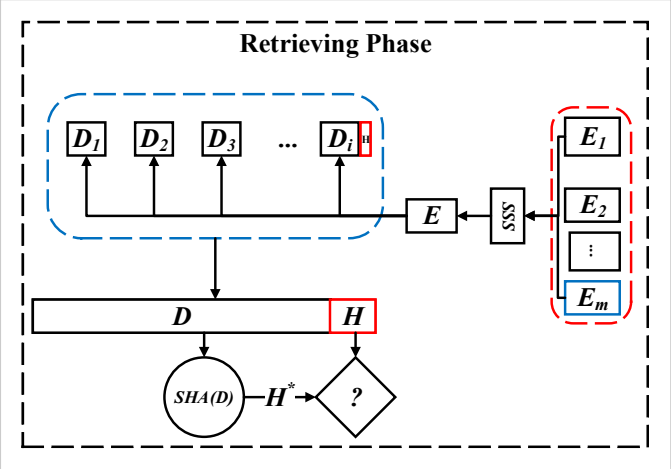

Figure 5. Schematic diagram of the Retrieving phase

polynomials equation, as shown in (1).

$$L(0) = \sum_{j=0}^{m-1} f(x_j) \prod_{\substack{q=0 \\ q \neq j}}^{m-1} \frac{x_q}{x_q - x_j}, \quad (1)$$

where $L(0)$ represents the retrieved key (E), and each point pair $(x, f(x_j))$ refers to one share.

Afterward, the healthcare data parts are collected from the storage locations, assembled according to their identifiers, and decrypted using the retrieved E-key. Moreover, one more step is needed to verify the integrity of the healthcare data. The secure hash algorithm is used to accomplish the last step, where the SHA-512 is used to compute the hash value (H^*) of the retrieved healthcare data D. The computed hash value (H^*) and the appended hash value (H) are compared to determine whether they are equal. If D is correctly gathered and not modified during transmission, the values of H and H^* are equal. Otherwise, D is corrupted and contains tampered contents. In the case of retrieving an individual record, the *SE* determines which data part the requested record belong to. Then, the data part block is decrypted to retrieve the required record.

V. Results and Discussion

The experiments were tested using a configurable experimental environment for High-Performance Computing (HPC) using Center for Computationally Assisted Science and Technology (CCAST) at the North Dakota State University. On CCAST, we reserved a cluster lease with 2-Intel Xeon processors 2.5GHz with 56 threads and 64GB of RAM. We tested the proposed design using a sample of synthesized data. The size of this sample is equal to *5GiB*.

A. Experimental analysis

To test the speed of the proposed design, we measured the elapsed time to collect and hash the test sample data. The results show that the proposed design requires 26.5 seconds to compute the hash value for the collected sample data. The time needed to collect the sample data parts and decrypts them is equal to 475.26 seconds. Overall, the total time to hash and collect the sample data parts shows the significant speed of our proposal over other designs in the literature, neglecting the time needed to collect the encryption key (E). Moreover, the decentralized approach that we proposed using the *SSS* algorithm provides a secure and authentic mechanism to store and monitor the healthcare data over the cloud.

B. Security Analysis

The security of the proposed design is expressed from the healthcare data owners perspective. For each security threat, we built a security model to verify the efficiency of the proposed design. This model includes the employed security functions (SSS and SHA-512) and experimental results. The security analyses results were deduced according to four deductions.

Deduction 1: The healthcare data parts are distributed over multiple locations and identified by a unique identifier for each part.

The first level of security is accomplished using the service entity, where the (*SE*) is the only one who is responsible for data summing according to their identifiers.

Deduction 2: The encryption key (E) is divided into shares and distributed over multiple authorized entities.

The second level of security is obtained using the SSS algorithm, where the decentralized encryption key scheme protects the encryption key (*E*) even if parts of the secret shares are exposed to an adversary.

Deduction 3: The healthcare data are integral and authentic, thanks to the *SHA-512*.

The third level of security is achieved by using the *SHA-512*. After retrieving the healthcare data, the *SHA-512* compression function is applied to make sure the retrieved data are tamper-free and gathered in the correct order.

Deduction 4: The level of trust between the healthcare data owners and the *CSPs* is increased.

The healthcare data owners make sure that the *CSP* has no access to the stored data. This is because the data are encrypted and the encryption key is distributed over multiple users. Moreover, the proposed design allows the healthcare data parts to be distributed over different *CSPs* which is supported by the data identifiers and the *SE*.

The employment of the SHA and SSS algorithms consolidates the healthcare data storage and sharing and increases the level of trust between the *CSP* and client. The proposed design accomplished the security requirements of data integrity and CSP prevention. Moreover, the use of SSS consolidates the proposed design against centralized control.

 ISBN: 978-1-61208-743-6

VI. Conclusions and Future Work

In this paper, a secure storage scheme for healthcare data was presented. The proposed design employed the *SSS* algorithm and *SHA-512* to provide the security requirements to the proposed design. The *SSS* is used to divide the encryption key into secret shares and distributes these shares on multiple authorized entities. Also, the *SHA-512* ensures that the healthcare data parts are tamper-free and collected in the correct order after retrieval. The proposed design is tested using a synthesized data sample using a high-performance computing environment. The results showed a significant speed in retrieving the healthcare data, and the security analysis provides the security proof of the proposed design.

In the future, further experiments will be conducted to include different samples. The security requirements will be extended to include the case of third-party auditor (*TPA*). Moreover, a data replication scheme over the cloud will be applied to the proposed design using division and replication of data in the cloud for optimal performance and security.

Acknowledgments

This publication was funded by a grant from the United States Government and the generous support of the American people through the United States Department of State and the United States Agency for International Development (USAID) under the Pakistan - U.S. Science & Technology Cooperation Program. The contents do not necessarily reflect the views of the United States Government.

Computing services, financial and administrative support from the North Dakota State University Center for Computationally Assisted Science and Technology (CCAST) and the Department of Energy through Grant No. DE-SC0001717 are gratefully acknowledged.

References

[1] C. Burghard, "Big data and analytics key to accountable care success," *IDC health insights*, pp. 1–9, 2012.

[2] J. G. Ronquillo, J. Erik Winterholler, K. Cwikla, R. Szymanski, and C. Levy, "Health it, hacking, and cybersecurity: national trends in data breaches of protected health information," *JAMIA Open*, vol. 1, no. 1, pp. 15–19, 2018.

[3] M. Ahmadi and N. Aslani, "Capabilities and advantages of cloud computing in the implementation of electronic health record," *Acta Informatica Medica*, vol. 26, no. 1, p. 24, 2018.

[4] K. Abouelmehdi, A. Beni-Hessane, and H. Khaloufi, "Big healthcare data: preserving security and privacy," *Journal of Big Data*, vol. 5, no. 1, p. 1, 2018.

[5] C. Tankard, "Big data security," *Network security*, vol. 2012, no. 7, pp. 5–8, 2012.

[6] W. Wilkowska and M. Ziefle, "Privacy and data security in e-health: Requirements from the users perspective," *Health informatics journal*, vol. 18, no. 3, pp. 191–201, 2012.

[7] R. Lu, H. Zhu, X. Liu, J. K. Liu, and J. Shao, "Toward efficient and privacy-preserving computing in big data era," *IEEE Network*, vol. 28, no. 4, pp. 46–50, 2014.

[8] Y. Li, K. Gai, L. Qiu, M. Qiu, and H. Zhao, "Intelligent cryptography approach for secure distributed big data storage in cloud computing," *Information Sciences*, vol. 387, pp. 103–115, 2017.

[9] Z. A. Al-Odat, S. K. Srinivasan, E. Al-Qtiemat, m. a. latha dubasi, and s. shuja, "Iot-based secure embedded scheme for insulin pump data acquisition and monitoring," in *The Third International Conference on Cyber-Technologies and Cyber-Systems*. IARIA, 2018, pp. 90–93.

[10] Z. Al-Odat, M. Ali, and S. U. Khan, "Mitigation and improving sha-1 standard using collision detection approach," in *2018 International Conference on Frontiers of Information Technology (FIT)*. IEEE, 2018, pp. 333–338.

[11] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979. [Online]. Available: http://doi.acm.org/10.1145/359168.359176

[12] I. B. Damgård, "A design principle for hash functions," in *Advances in Cryptology — CRYPTO' 89 Proceedings*, G. Brassard, Ed. New York, NY: Springer New York, 1990, pp. 416–427.

[13] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche, "Sponge functions," in *ECRYPT hash workshop*, vol. 2007, no. 9. Citeseer, 2007, pp. 1–93.

[14] F. PUB, "Secure hash standard (shs)," *FIPS PUB 180*, vol. 4, pp. 1–27, 2012.

[15] B. L. Filkins *et al.*, "Privacy and security in the era of digital health: what should translational researchers know and do about it?" *American journal of translational research*, vol. 8, no. 3, p. 1560, 2016.

[16] Y. X. Yan, L. Wu, W. Y. Xu, H. Wang, and Z. M. Liu, "Integrity audit of shared cloud data with identity tracking," *Security and Communication Networks*, vol. 2019, pp. 1–11, 2019, doi.org/10.1155/2019/1354346.

[17] Y. Zhang, P. Lang, D. Zheng, M. Yang, and R. Guo, "A secure and privacy-aware smart health system with secret key leakage resilience," *Security and Communication Networks*, vol. 2018, pp. 1–13, 2018.

[18] R. Charanya, S. Nithya, and N. Manikandan, "Attribute based encryption for secure sharing of e-health data," in *Materials Science and Engineering Conference Series*, vol. 263, no. 4, 2017, p. 042030.

[19] B. Shen, J. Guo, and Y. Yang, "Medchain: Efficient healthcare data sharing via blockchain," *Applied Sciences*, vol. 9, no. 6, p. 1207, 2019.

[20] M. Li, S. Yu, K. Ren, and W. Lou, "Securing personal health records in cloud computing: Patient-centric and fine-grained data access control in multi-owner settings," in *International conference on security and privacy in communication systems*. Springer, 2010, pp. 89–106.

[21] T.-S. Chen *et al.*, "Secure dynamic access control scheme of phr in cloud computing," *Journal of medical systems*, vol. 36, no. 6, pp. 4005–4020, 2012.

[22] A. Babatunde, A. Taiwo, and E. Dada, "Information security in health care centre using cryptography and steganography," *arXiv preprint arXiv:1803.05593*, 2018.

[23] J. Vora *et al.*, "Ensuring privacy and security in e-health records," in *2018 International Conference on Computer, Information and Telecommunication Systems (CITS)*. IEEE, 2018, pp. 1–5.

 ISBN: 978-1-61208-743-6